Eerno Niemenmaa

Computing Nash equilibrium in a telecommunication market model with different cognitive radio scenarios

School of Science

Thesis submitted for examination for the degree of Master of Science in Technology.

Espoo October 25, 2013

Thesis supervisor:

Prof. Harri Ehtamo

Thesis instructor:

D.Sc. Kimmo Berg

Aalto University
School of Science

AALTO UNIVERSITY
SCHOOL OF SCIENCE

ABSTRACT OF THE
MASTER'S THESIS

Author: Eerno Niemenmaa

Title: Computing Nash equilibrium in a telecommunication market model with different cognitive radio scenarios

Date: October 25, 2013 | Language: English | Number of pages:3+46

Department of Mathematics and Systems Analysis

Professorship: Applied mathematics | Code: Mat-2

Supervisor: Prof. Harri Ehtamo

Instructor: D.Sc. Kimmo Berg

In this thesis we are investigating how the different cognitive radio scenarios affect the operators, the customers and the government. We have constructed a telecommunication market model to find out the consequences. We have constructed a two-stage game based on this model. In the first stage operators buy capacity. That means licensing spectrum and maintaining the network technology. In the second stage the operators set prices to their service. The special properties in our game are limited availability of capacity and uncertainty of demand. The limited availability of capacity is simulated by using a quadratic cost function. The uncertainty of demand is generated by adding a random variable to the demand function. The main theme of this thesis is comparing different scenarios to each other. For each scenario we developed functions to represent the additional capacity. Finally we compare the results by using indexes created to represent the utility of different sides.

Solving the game is started by using analytical methods with general number of players. However, the complexity of dependencies prevents solving game analytically. We continue with numerical methods. With numerical methods we primarily are searching pure strategy Nash equilibrium. It turns out that pure strategy equilibrium does not exist generally. To solve mixed strategy equilibrium, we form a normal form game to region that our algorithm keeps oscillating and solve its mixed strategy equilibrium.

According to the results of this thesis, one of the suggested changes in rules would be equally profitable for operators, but remarkably better by the government's and the customers' point of view.

Keywords: Cognitive radio, Telecommunication market model, Two-stage game, Normal-form game, Oligopoly game, Game theory, Mixed strategy equilibrium

AALTO-YLIOPISTO
PERUSTIETEIDEN KORKEAKOULU

DIPLOMITYÖN
TIIVISTELMÄ

Tekijä: Eerno Niemenmaa

Työn nimi: Telekommunikaatiomarkkinoita kuvaavan mallin Nashin tasapainon laskeminen erilaisilla kognitiiviradion skenaarioilla

Päivämäärä: October 25, 2013 Kieli: Englanti Sivumäärä:3+46

Matematiikan ja systeemianalyysin laitos

Professuuri: Sovellettu matematiikka Koodi: Mat-2

Valvoja: Prof. Harri Ehtamo

Ohjaaja: D.Sc. Kimmo Berg

Tässä työssä tutkittiin kognitiivisen radion erilaisten toteutusskenaarioiden vaikutuksia operaattoreiden, asiakkaiden sekä valtion kannalta. Työssä rakennettiin malli kuvaamaan telekommunikaatiomarkkinoita. Mallin pohjalta tehdään peli, joka koostuu kahdesta vaiheesta. Ensimmäisessä vaiheessa operaattorit hankkivat kapasiteettia, eli ostavat lisenssin spektriin sekä huoltavat verkkoa ylläpitävää tekniikkaa. Toisessa vaiheessa operaattorit hinnoittelevat palvelunsa. Rakentamassamme pelissä erikoispiirteinä ovat kapasiteetin rajallinen saatavuus sekä kysynnän epätarkka ennustettavuus. Kapasiteetin rajallisuutta simuloidaan sen neliöllisellä kustannusfunktiolla. Liian tarkan ennustettavuuden poistamiseksi kysyntäfunktioon on lisätty satunnaismuuttuja. Työn tarkoitus on erilaisten sääntöjen vertailu keskenään. Erilaisia skenaarioita varten on rakennettu omat funktiot kuvaamaan lisäkapasiteetin saamista. Lopuksi tuloksia vertaillaan eri osapuolien näkökulmista näiden tyytyväisyyttä kuvaamaan kehitettyjen indeksien avulla.
Työssä pelin ratkaisu aloitetaan analyyttisillä menetelmillä yleisellä pelaajamäärällä. Ongelmaksi kuitenkin muodostuu liian monimutkaiset riippuvuudet, jotka estävät pelin analyyttisen ratkaisemisen. Pelin tutkimista jatketaan numeerisilla menetelmillä. Numeerisella menetelmällä haetaan ensisijaisesti puhtaista strategioista koostuvaa Nashin tasapainoa. Osoittautuu kuitenkin, että aina puhtaiden strategioiden tasapainoa ei ole olemassa. Sekastrategiatasapainon ratkaisemiseksi muodostamme matriisipelin alueelle, johon algoritmimme jää kiertämään kehää. Lopuksi ratkaistaan toisella algoritmilla sen sekastrategiatasapaino.
Työn tulosten perusteella toinen ehdotetuista sääntömuutoksista olisi operaattoreiden tuottojen kannalta nykytilanteen kanssa yhtä hyvä, mutta valtion ja asiakkaiden kannalta selvästi nykytilannetta parempi.

Avainsanat: Kognitiivinen radio, Telekommunikaatiomarkkinamalli, Kaksivaiheinen peli, Matriisipeli, Oligopolipeli, Peliteoria, Sekastrategiatasapaino

Acknowledgements

This thesis has been done at the department of Mathematics and System Analysis in the Aalto University School of Science. I would like to thank my instructor Kimmo Berg for his insight and guidance during the research period of this thesis. My supervisor, Harri Ehtamo, I would like to thank for his patience and his hard work helping me to finish this thesis. I also want to thank all my co-workers at the department for their support and interesting conversations.

I also would like to thank my family and friends for their support in this thesis and studies overall.

Espoo, October 25, 2013

Eerno Niemenmaa

Contents

1 Introduction

Telecommunication business has been growing within a few decades from few devices into huge market. These days in developed countries almost everyone has at least a telephone and many have even more devices using radio wave technology. As the market keeps growing, the providers face new challenges. They need to satisfy the needs of the customers. All these devices need a little piece of bandwidth to work properly. As there is only limited amount of bandwidth available, we will sooner or later face difficulties. Running out of frequencies actually happens already for example in big public events.

The spectrum bandwidths are currently being licensed to different purposes. Some frequencies are licensed to telecommunication operators and TV channels. There are also some frequencies reserved for example for military use and for authority usage such as police. Meanwhile telecommunication frequencies are sometimes overloaded, other frequencies might be not in use at all. In some cases there might be even situations where some operators' frequencies are overloaded while some other operators still have bandwidth unused.

Motivated from this situation, J. Mitola [14] has invented the concept of cognitive radio. The idea is that if device's primarily used frequencies are overloaded, it could use other frequencies instead. There has been promising research to solve the technical difficulties in these devices. When this technology becomes reality the licensing system of the spectrum must be renewed. These days there are auctions where every telecommunication operator buys their own licensed bandwidths. This could be one possible way to carry on, but it does not receive the benefits from the cognitive radio technology. Another way would be to let operators license their own bandwidths, but in addition there would be a common bandwidth. In case some operator runs out of bandwidth, they could use these common frequencies. Yet another way would be that firms license their own bandwidths, but whenever one runs out of frequencies and another still has some left, this operator can borrow spectrum from the others.

It is totally possible to use any of the rules mentioned above. To make the decision between them we need to think their consequences. In this thesis we are investigating what kinds of behavior those different rules would lead to. We construct a market model for each rule. We consider market model as a game between operators. Then our aim is to solve the Nash equilibrium of the game. To draw concequences we use analytical methods as far as we can. Nevertheless, the solving of the problem turns out to be very challenging even numerically.

2 Literature Review

2.1 Cognitive radio

Concept of cognitive radio was first introduced by J. Mitola in [14]. There are some books dealing with versatile subjects related to cognitive radio. Such books are [15] written by B. Fette and [16] written by L. Barlemann and S. Mangold. Some subjects to mention there are radio spectrum regulation and usage, research results, history of cognitive radio, communication policy as well as technologies and some game theoretical things and many other things related to the project.

There have been many studies in some countries confirming the observation that radio spectrum was inefficiently utilized. Some of these studies are represented in [17] by D. Cabric, S. Mishira and R. Brodersen, [18] by V.Valenta and R. Maršálek and [19] by T. Weiss and F. Jondral.

There are still lots of research going on with cognitive radio topics. One essential topic is the management of licence auction. In [20] K. Berg, H. Ehtamo, M. Uusitalo and C. Wijting have studied the current auction system and stated some lacks in it. They introduced a new taxonomy to replace the current auction system.

2.2 Oligopoly theory

Oligopoly theory is a field of science which studies markets with small number of players. This is a topic of huge interest and can be interpretted in a pure mathematical point of view or as tools for economists to explain the laws of market. The basic concepts are pretty much the same for both of the previous view points. In [4] X. Vives begins with basic consepts of game theory and then gives a vide overview on oligopoly theory. His book covers concepts like classical market models such as Cournot's and Bertrand's models as well as more complicated systems. In R. Gibbon's book [1] there are lots of game theoretic basic consepts and the basics of oligopoly theory explained throughout in a compact package. In [3] D. Fudenberg and J. Tirole gives more extensive overview on game theory.

Games tends to become unsolvable analytically when we have even a bit more complicated games than the basic examples. Thus, there are articles conserning also the numerical methods to handle games with a small amount of players. Some numerical methods to find Nash equilibrium in oligopoly games are introduced in T. Basar's article [6] and J. Krawczyk's and S. Uryasev's article [7]. The main results are the convergence results as well as the algorithms themself.

2.3 Solving mixed strategy equilibria

In a game there are players who play the game by taking actions. Each player has a strategy which defines the process of taking actions in each possible situation in the game given the information provided. Strategy is a function of game history which also depends explicitly on the situation at hand. The strategy may also explicitly depend on random variables, e.g., in a particular situation, take action a with probability p and action b with probability $1 - p$. Strategies which include random variables are called mixed strategies. If we have a continuous interval of possible actions in some part of a game, we say that we have a continuous strategy set. Mixed strategies in continuous strategy sets are any probability distributions on those sets.

The standard solution concept of a game is the Nash equilibrium. It means a profile of strategies which satisfies condition that any player alone does not benefit by deviating from his strategy in this profile. Nash equilibrium concept is extended to consider also mixed strategies by considering the expected payoffs.

Solving mixed strategy equilibrium of a one-stage game with continuous strategy sets is basically finding probability distributions such that if each player i is using his distribution to define his action, any player j should not benefit by deviating from using his distribution. This problem has too many degrees of freedom to be solved numerically. One way to approach this problem is to let probability distributions be of a particular function type with a few parameters and solve the problem in this subspace.

One way of choosing the subspace of probability distributions with low degree of freedom is to discretize the sets of actions available for each player. Then we end up with a finite one-stage game which can be represented as a normal-form game. In [8] R. Porter, E. Nudelman and Y. Shoham have developed an algorithm which solves mixed strategy equilibria of normal form games. They showed that their algorithm finds Nash equilibrium in any normal-form game and in many cases faster than earlier published algorithms.

2.4 Telecommunication market model analysis

Many papers have been written on telecommunication business. Differences arise from that there are some ways to define the relations between products' prices, demands and profits of firms. Another thing is that this market obviously has two layers. First operators have to buy capacity, and after that they still have the price game to play. In [10] J. Jia and Q. Zhang have researched non-cooperative two-stage duopoly market without any stochastic involved. They have shown really nice analytical results that their duopoly market has unique Nash equilibrium and they have algorithms which they have shown to converge to the equilibrium. To reduce non-rational behaviour that operators would really have full information how the demand develops, they have also studied game where the price game is a dynamic game where operators changes their prices according to the best response dynamic.

When the amount of players increases, the complexity of the problem increases. Analytical results are much harder to deal with and uniqueness of Nash equilibrium is no longer so obvious. In [11] Y. Xu, J. Liu and D. Chiu have studied two-stage non-cooperative oligopoly game where they have much more complex relations between prices, demands and profits. They also have researched an additional penalty function to represent the ability to use non-licensed spectrum, but with significantly higher price. They have used same kind of dynamics as [10].

As telecommunication markets have pretty low marginal costs, there might be a lot potential profits achieved for operators by creating coalitions. In [9] A. Klementtilä has researched how coalitions affect in a market and differences between coalitions.

3 Nash equilibrium

The purpose in this thesis is to build a game theoretic model to study the equilibrium of a market of a telecommunication business with a couple of different rules implied to the market.

To define the *game*, we first need some notations and functions to be defined. In games there is a set of *players*. Amount of players is denoted by n. Let p denote a profile vector $(p_1, \dots, p_n)$, where p_i is an item, e.g. price of a product, for player i. We also denote by $p_{-i} := (p_1, \dots, p_{i-1}, p_{i+1}, \dots, p_n)$ a vector of items without player i 's item included.

The players play a game by taking *actions*. We call *strategy* a rule which determines player's actions at any stage of the game. We denote the strategy for player i by s_i and assume that $s_i \in \mathbb{S}_i$, which we call the *strategy set* for player i. For each $s = (s_1, \dots, s_n) \in \mathbb{S}_1 \times \cdots \times \mathbb{S}_n$, we define a *payoff* for player i by $\pi_i(s_1, \dots, s_n)$. Thus π_i is a mapping

$$\pi_i : \mathbb{S}_1 \times \cdots \times \mathbb{S}_n \to \mathbb{R},$$

which we call a payoff function for player i.

Definition: Game. Game specifies the players' strategy sets $\mathbb{S}_1, \dots, \mathbb{S}_n$ and their payoff functions $\pi_1, \dots, \pi_n$. We denote this game by $G := \{\mathbb{S}_1, \dots, \mathbb{S}_n, \pi_1, \dots, \pi_n\}$.

There are several typical ways to represent games graphically. The *normal-form* game means that a game is represented in a matrix form. There are also many kinds of games in terms of information provided to the players during the game. We focus on games which has *complete information*.

Definition: Complete information. A game G is said to have complete information if every player knows strategy sets and payoff functions in the game.

In game theory players are usually assumed to act rationally. The main goal is to find some equilibrium solutions to games. With rational players these equilibria should be the outcomes of the game. The most common equilibrium is called *Nash equilibrium* named after famous game theorist John Nash [5]. Players are playing strategies which form the Nash equilibrium if none of them are willing to deviate from his equilibrium strategy when other players play their equilibrium strategy.

Definition: Nash equilibrium (NE).

A profile of strategies $s^* = (s_1^*, \dots, s_n^*) \in \mathbb{S}$ forms a Nash Equilibrium if

$$\pi_i(s_{-i}^*, s_i^*) \geq \pi_i(s_{-i}^*, s_i), \quad \forall s_i \in \mathbb{S}_i,\ \forall i \in \{1, \dots, n\}.$$

Example: We introduce a famous game called *prisoner's dilemma.* There are two players who have committed a crime and are now being interrogated separately. If they both deny the crime, they both shall be free and have payoff 0. If they both admit the crime, they shall have 3 years in prison each. So this will result payoff of -3. If one of them admits and another denies, the one who admits shall be free and get a reward and the other shall have 4 years in prison. Their payoffs are 1 and -4. This game is represented as a normal-form game in Figure (1).

		2	
		A_2	D_2
1	A_1	-3,-3	1,-4
	D_1	-4,1	0,0

Figure 1: Normal-form representation of prisoners dilemma.

Now player i can choose whether to admit (A_i) or deny (D_i). Both players see that what ever the other chooses, his own payoff will be higher when he chooses to admit. So the outcome of the game is that both admit. This is the unique Nash equilibrium of the game. The inequalities in the definitioin of Nash equilibrium are

$$\pi_1(A_1, A_2) = -3 \geq -4 = \pi_1(D_1, A_2)$$
$$\pi_2(A_1, A_2) = -3 \geq -4 = \pi_2(A_1, D_2).$$

There is a paradox hidden in this game. There is a payoff pair $(0, 0)$ which gives a better result for both players than the Nash equilibrium payoff. Nevertheless, this pair cannot be obtained without coordination. Modeling this leads to the most interesting applications in game theory.

Strategies can be either deterministic or include probability distributions over actions. Deterministic strategies are called *pure strategies* and strategies with probability distributions over actions are called *mixed strategies.* In the definitioin of

Nash equilibrium we must use expected values in inequalities instead of direct payoff functions.

Example: In this example we introduce another famous game, called *matching pennies*. There are two players who simultaneously decide whether to put coin in their hand heads (H) or tales (T) upwards. Then they open their hands and if the top sides match, player 1 wins and othervise player 2 wins. The game is represented in Figure 2.

		2: H	2: T
1	H	1,-1	-1,1
1	T	-1,1	1,-1

Figure 2: Normal-form representation of matching pennies.

Now there is no profile of actions satisfy the Nash equilibrium condition. However there is a mixed strategy equilibium (s_1^*, s_2^*). As there is only two options of actions for each player, the equilibium mixed strategy must be of the type that player 1 plays (H) with probability p and (T) with probability $1-p$, and player 2 plays (H) with probability q and (T) with probability $1-q$. To satisfy the Nash equilibrium condition, player 1 must be *indifferent* between choosing (H) or (T) when player 2 plays his equilibrium strategy. This means that his expected payoff will be the same by playing either of those actions. Also player 2 must be indifferent when player 1 plays his equilibrium strategy. These two conditions are enought to determine the unknow probabilities p and q.

$$\begin{aligned} \mathbb{E}\pi_1(H, s_2^*) = 1q + (-1)(1-q) = (-1)q + 1(1-q) = \mathbb{E}\pi_1(T, s_2^*) \\ \mathbb{E}\pi_2(s_1^*, H) = (-1)p + 1(1-p) = 1p + (-1)(1-p) = \mathbb{E}\pi_2(s_1^*, T) \end{aligned} \tag{1}$$

The only solution for equations 1 is $(p, q) = (\frac{1}{2}, \frac{1}{2})$ and the expected payoffs are $(0, 0)$. This profile of strategies satisfies Nash equilibrium conditions.

There are two kinds of games in terms of the order in which the actions are taken.

Definition: static game. A game is *static* if all actions are taken simultaneously.

Definition: dynamic game. A game is *dynamic*, if there are *stages* in which one or more players take their actions.

The model we are studying in this thesis is a *two-stage game*. This means that there are two stages in which players choose their actions simultaneously.

In some games, between the stages players might have an opportunity to observe what other players have chosen in earlier stages. Whether this is allowed or not has a huge impact on the solution of the game. If it is allowed, we say the game has *perfect information*.

Definition: perfect information. Game has perfect information if at each time player makes a decision between actions in the game he knows the full history of the game thus far.

We are investigating only the case of perfect information.

In our study a two-stage game is essential. Thus, we want to introduce the concept of a *subgame*. In our case subgame simply means the game left to play given actions in the first stage.

Example: We have a two-stage complete and perfect information game with two players. On the first stage players simultaneously choose between heads (H) and tales (T). On the second stage they are informed about the outcome of the first stage and they again choose actions simultaneously. Player 1 chooses between up (U) and down (D) and player 2 chooses between left (L) and right (R). The outcome of the first stage determines the payoffs of each outcome in the second stage. The game is represented in Figure (3). In the figure, the Nash equilibria of subgames are circulated. Thus, if the first stage is played such that a particular subgame is reached, the payoff will be the one circulated.

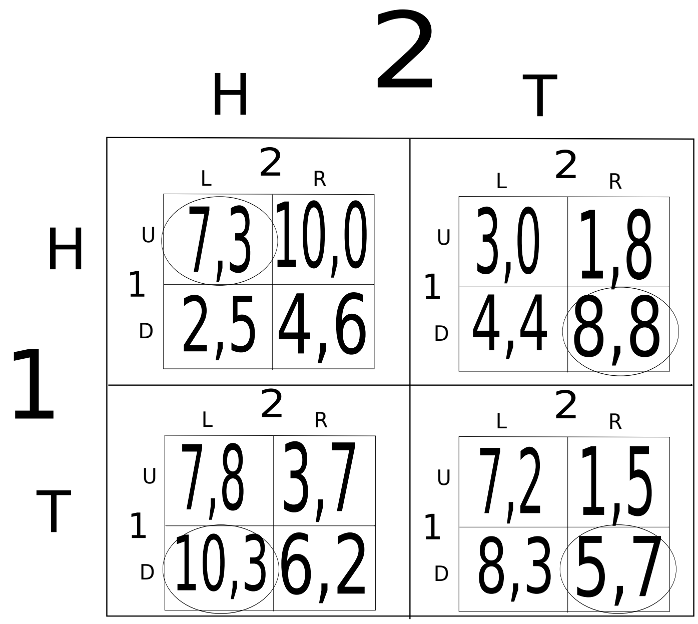

Figure 3: Two-stage game with complete and perfect information in normal-form representation.

In Figure (4) we have replaced each subgame by its outcome. Now we have a one-stage game which we can solve as such. The unique Nash equilibrium is again circulated. So we conclude that this is Nash equilibrium outcome of the two-stage game.

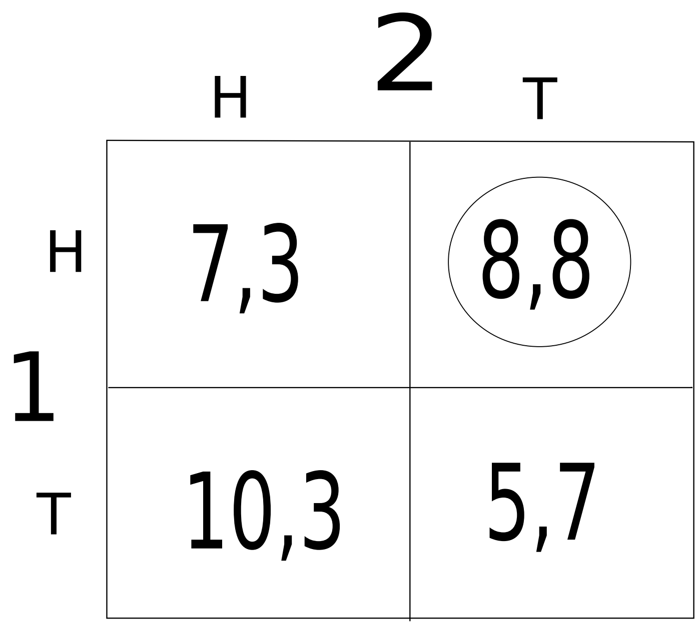

Figure 4: Normal-form representation of prisoners dilemma.

The previous example was solved using the method called *backwards induction.* This means that in N-stage game with every possible profile of actions in the $N-1$ first stages we solve the equilibrium in the last stage as a subgame. When we have equilibria solved for each of these subgames we can simply replace the subgames by their outcomes in $(N-1)$th stage. By repeating the procedure by working backwards the game finally reduces to a one-stage game.

Definition: subgame perfect equilibrium. A Nash equilibrium is *subgame-perfect* if the players' strategies constitute a Nash equilibrium in every subgame.

Sometimes we dont have a unique Nash equilibrium. This kind of situation is demonstrated in the following example.

Example: There are two firms as players and both can either buy high (H) or low (L) amount of capacity. In the second stage they can either decide to set prices low (L) or high (H). The payoffs of the game are represented in Figure 5.

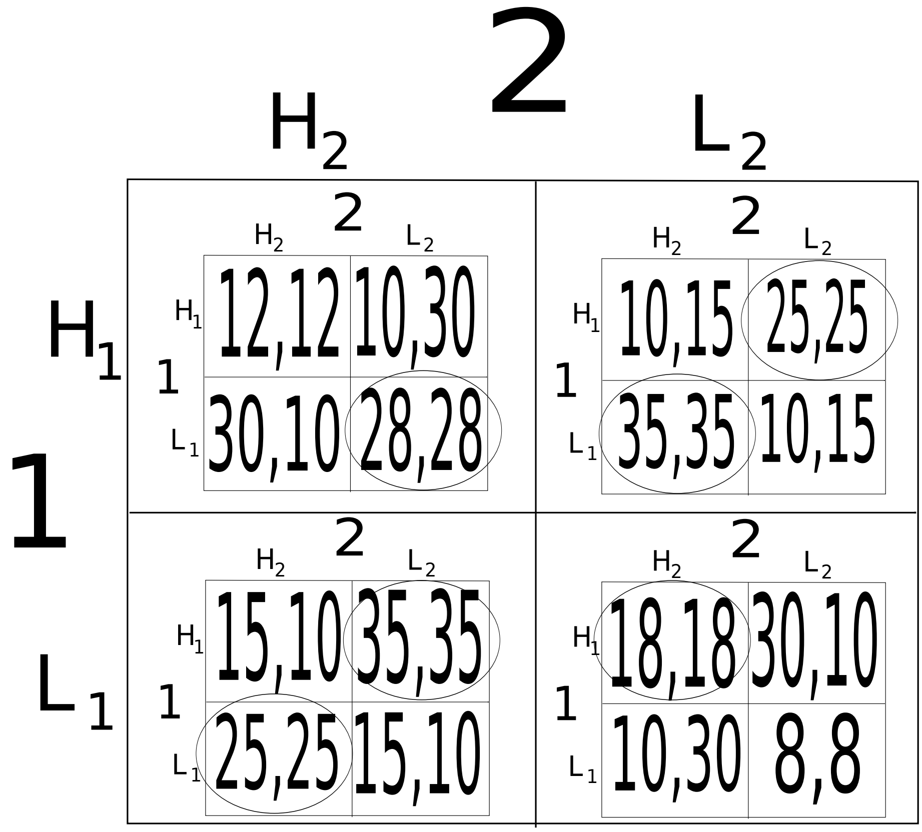

Figure 5: Four subgames. Pure strategy Nash equilibria of each subgame are circulated.

In every subgame the Nash equilibria are circulated. In second and third subgame there are two Nash equilibria, but it seems pretty clear that the one with higher payoffs will be played. Assuming this we simply replace the subgames by its Nash equilibrium outcome and we will receive a normal-form game with one stage and the payoffs are the outcomes of the subgames. The game reduce to a one-stage game and it is represented in Figure 6.

1 \ 2	H_2	L_2
H_1	28,28	35,35
L_1	35,35	18,18

Figure 6: Game reduce to its first decision node. Each subgame is replaced by its subgame Nash equilibrium outcome. Nash equilibria are circulated.

Now we have again circulated the Nash equilibria of this new one-stage game. Given this logic of solving the game by backwards induction our Nash equilibria of this game are for player 1 to play high capacity in the first stage and low price in the second stage and for player 2 to play low capacity and high price, respectively. Another Nash equilibrium is for player 2 to play high capacity and low price and for player 1 to play low capacity and high price.

In the previous game, the subgames which have two pure strategy Nash equilibria, also have mixed strategy equilibria. That is for the player who played high capacity in the first stage, say player 1, plays high price (low price) with probability $\frac{2}{3}$ $(\frac{1}{3})$ in the second stage. Player 2 plays high price (low price) with probability $\frac{3}{8}$ $(\frac{5}{8})$ in the second stage. In this equilibrium the expected payoffs are $\frac{155}{8}$ and $\frac{65}{3}$ for player 1 and player 2, respectively. This is remarkably lower than either of the pure strategy equilibria. In general, when we have multiple Nash equilibria in a game their outcomes could differ dramatically from equilibrium to equilibrium; in such cases it is difficult to decide what equilibrium to play.

Next, we introduce *the best response function* for player i denoted by BR_i.

Definition: Best response function. The best response function of player i is defined by

$$BR_i(s_{-i}) := arg \max_{s'_i \in \mathbb{S}_i} \pi_i(s_{-i}, s'_i).$$

where (s_{-i}, s'_i) means profile of strategies where player i 's strategy s_i is replaced by s'_i.

This function defines the strategy that maximizes the payoff of player i when the other players play s_{-i}.

If the maximum in the definition is not unique, the best response function is defined to be set valued. If a profile of strategies $s^* = (s_1^*, \ldots s_n^*) \in \mathbb{S}$ satisfies the condition

$$s_i^* \in BR_i(s_{-i}^*), \quad \forall i \in \{1, \ldots, n\},$$

then by definition, s^* is a Nash equilibrium of the game. Mathematically, s^* is a fixed point of the system of all players' best response functions. From this we get a typical fixed point iteration to find a Nash equilibrium. We return to this later in the section of algorithms.

4 Economic theory

4.1 Monopoly theory

In this subsection we are considering *monopolies*, market forms with one firm involved. In section 4.2 we shall study two competing firms and take the game theoretic point of view. To study what kind of payoff functions we should have in our model, we need to define some economic terms.

There are two types of model to descripe the competition between the firms. According to *the Cournot model*, each firm chooses how much to produce its product and the price of the product will be determined from that. In this thesis we are using another approach which is called *the Bertrand model*. In Bertrand model each firm chooses what is the price of its product. The *demand* of the product is then determined by the price.

First, we need the *demand function*. Demand function defines how many products are sold with a given price p. For simplicity, we use in this thesis linear demand function. In one firm case demand function is

$$d(p) = a - bp,$$

where p is the price of the product and a and b are positive constants. Constant b tells the marginal increment of demand when price increases one unit. A very important aspect of demand function is its *elasticity*. Elasticity is defined as

$$E(p) = \frac{\partial d(p)}{\partial p} \frac{p}{d(p)},$$

where $d(p)$ is the demand function. Elasticity tells how sensitive the demand is for changes in price. Constant b in demand function is closely related to elasticity. However, elasticity is not a constant even with linear demand function because it represents a relation between price and demand relatively. Thus, calculating elasticity yields

$$E(p) = \frac{\Delta d(p)}{\Delta p} \frac{p}{d(p)} = -b \frac{p}{a - bp}.$$

For example, in telecommunication business the parameters a and b of the model are such that elasticity is typically in interval $[1.3\ ,\ 3]$. If the value of elasticity is far away from this interval, we will conclude that the parameter values of the model are wrong. For more about economic models, see [12] written by A. Mas-Colell, M. Whinston and J. Green.

So firm chooses the price, which defines the demand. Firm can not sell more than that amount of products, but it may happen that he is not able to sell even that much. Thus, we define *quantity* q to represent the amount of products actually being sold. We have a natural restrinction that the quantity of products being sold does not exceed the demand, i.e., $q \leq d$. If no other constraints are active, those will be equal. Anyway, in this thesis we have constraints and we need both consepts, demand and quantity, of sold products. We return to these constraints later in the chapter of capacity sharing rules.

The target for the firm is to maximize its profit, which defines the payoff function in games between firms. For this we need to define the payoff function π of this model. Production costs are commonly decomposed into two parts, *marginal cost* and *fixed cost*. Marginal cost tells the marginal increment of costs when production is increased by one unit. For simplicity, it is taken as a constant. Fixed cost is also kept as a constant. It normalizes the amount of costs to the right level. Fixed cost is the costs of production that are independent of the quantity of production. So the payoff function is

$$\pi(p) = (p - c_M)d(p) - c_F,$$

where p is the price, $d(p)$ is the demand function, c_M is the marginal cost and c_F is the fixed cost.

For this simple game we calculate the price for maximal payoff. Putting the linear demand function into the payoff function we get

$$\pi(p) = (p - c_M)d(p) - c_F = (p - c_M)(a - bp) - c_F.$$

We find maximum of this function by using the first order optimality condition

$$0 = \frac{\partial \pi(p)}{\partial p}\bigg|_{p=p^*} = a - bp^* - b(p^* - c_M),$$

from which we solve the optimal price p^*:

$$p^* = \frac{a + c_M b}{2b}.$$

We see that the optimal price is dependent on the demand function's parameters a and b as well as the marginal cost c_M, but not on fixed cost c_F. This is reasonable since fixed cost plays just a role of a normalizer.

4.2 Duopoly theory

In monopoly the one firm has the right to choose the price as he want and customers always have to either buy the product or not. If there are more firms in the same market, there will be competition. If there are two firms in a market, the market is called *duopoly* and for several firms it is called *oligopoly*. If the amount of firms increases so much that a single firm has no market power, i.e., his action alone does not effect on other firms' payoffs, the market has *perfect competition*, and can be considered as a market with infinite number of firms. As we have continuum in spaces of prices, demands and quantities of sold products, there are no problems with this interpretation. In perfect competition case with identical products the demand functions acts such that no products are sold if price is greater than the marginal cost and no firm will make profit. The reason for this is that if there were a firm selling with higher price than the others, his demand would be zero. Thus, no firm wants to be the one that sells with higher price than the others until the prices reach the amount of the marginal cost. In this thesis we concentrate on games with two or three firms.

In duopoly market, we use the Bertrand duopoly model. Again we consider market as a game and the firms are the players of this game. The demand of each player is affected not only by his own price but also by the other player's price. We still want to keep things relatively simple, so we have a linear demand function. Thus, the demand function in this duopoly market for each player is

$$d_i(p) = a_i - b_i p_i + c_i p_{-i}, \quad i = 1, 2$$

where p_i is price for player i 's product and $p = (p_1, p_2)$ is a profile of prices, a_i, b_i and c_i are constants. Constants a_i and b_i are same as in monopoly case. We introduce another constant c_i which tells how much player i 's demand is affected by unit increment of another player's price. In economic theory this constant can be either negative or positive. If it is negative, products are called *complementary*. In this case increase in one player's demand actually increases the demand of another player's product. Examples of this are telephones and cells for them or tables and chairs. If constants c_i are positive the products are called *substitutes*. Usually when players have similar kind of products, customers have to make a decision between the two products. Then whenever another firm lowers price, customers are more likely to buy products from that player and not from the other. It is also possible that c_i:s are zeros. In this case we call products *independent* and form two separate markets. In our thesis we are concentrating in situation where products are highly substitutes since each customer obviously chooses at most one operator at a time.

In Bertrand duopoly model, the dependencies are easy enough for us to find the unique Nash equilibrium. The payoff functions for each player are the same as they were in monopoly case. For each player we put the demand functions to the payoff functions

$$\begin{aligned}\pi_i(p) =&(p_i - c_M)d_i(p) - c_F\\ =&(p_i - c_M)(a_i - b_i p_i + c_i p_{-i}) - c_F, \quad i = 1, 2.\end{aligned}$$

By using the first order optimality condition for both players payoff functions we get

$$\begin{aligned}0 =& \left.\frac{\partial \pi_i(p)}{\partial p_i}\right|_{p_i = p_i^*,\ p_{-i}=p_{-i}^*}\\ =&a_i - b_i p_i^* + c_i p_{-i}^* - b_i(p_i^* - c_M), \quad i = 1, 2.\end{aligned}$$

This is a linear system of two equations and two unknowns, so it is easy to solve. If we assume symmetric situation, meaning the same demand function parameters for both players, we get

$$p_i^* = \frac{a + bc_M}{2b - c}$$

as a solution. If products are independent ($c = 0$) the solution is equal to the monopoly case as one could expected. If the products are substitutes the prices will be higher than in monopoly case and if products are complementary, prices will be lower.

4.3 Oligopoly theory

In oligopoly case demand function follows the same ideas already applied in monopoly and duopoly situations. We want demand to be a linear function of prices of each player. To prevent the amount of parameters of the model from growing up too fast, we will have one constant for each player describing the effect of all other players' increments in price

$$d_i(p) = a_i - b_i p_i + c_i \sum p_{-i}, \quad \forall i \in \{1, \dots, n\} \tag{2}$$

where $\sum p_{-i}$ is a shorthand notation for sum of prices not including p_i. Thus, c_i represents the increment of demand of player i whenever other players increase their prices in total of one unit. Payoff functions are exactly the same as in previous cases. Putting demand functions to the payoff functions yields quadratic functions and by using the first order optimality condition we again get a linear set of equations with n equations and n unknowns, so this can be solved as the previous case.

We have replaced constants describing how much player i 's demand will depend on another player j:s demand by just one constant c_i. Thus, we have implicitly assumed that this relation is relatively same between player i and any other player, too. This implies that all products must be substitutes pairwise since otherwise there would be problems. As described earlier, in this thesis we have products which are heavily substitutes to each other, so this assumption causes no harm.

5 Capacity sharing game

5.1 Model and stochastics

In this chapter, we construct our model to represent telecommunication market as a game. Operators are considered as the players of this game. We start by considering the special features of the payoff function, which is introduced in the previous chapter. We recall that the payoff function for player i in general is

$$\pi_i(p) = (p_i - c_M)q_i(p) - c_F,$$

where we used q_i instead of d_i to emphasize that payoff is calculated from the quantity of actually sold products, not from the demand.

In telecommunication business players have to decide beforehand which frequency bands they are going to buy from license auctions and after licencing they have to build and maintain the network. We define *capacity* to represent the amount of units of customers that a network can serve. Costs related to licencing and maintaining network are clearly fixed costs in the payoff function since they are independent of how many customers are actually buying their product. On the other hand, the marginal cost from serving one more unit of customers is near to zero compared to the price of the product. For these reasons we approximate c_M to be zero and c_F to be a function of capacity. We could use a linear function of capacity to represent the fixed cost. However, the primary problem in this thesis is that the capacity is running out so we need a function where marginal cost of buying capacity grows as the amount grows. To keep things simple, we use a quadratic function in terms of capacity to represent the fixed cost. So payoff function for player i is

$$\pi_i(p, k) = p_i q_i(p, k) - g k_i^2,$$

where g is a positive constant and k_i is the amount of capacity of player i. We have also added k to be argument in q_i since it defines restrinctions as an upper bound for q_i.

Next, we consider the demand function of our model. We use the demand function defined in last chapter for several players. Once we have constraints determined by the capacity, there will be a really sharp edge in the payoff function at the point where the capacity runs out. It is obvious that no player is going to price his product too cheap such that his frequencies would become overloaded, so it might be optimal to set prices such that the band is exactly full. It is obviously not realistic to expect the demands to be exactly those given by the function (2). To avoid this unrealistic behavior we have added a random variable to the demand function, so players have to take into account some uncertainty. Thus, the demand function for player i is

$$d_i(p) = a_i - b_i p_i + c_i \sum p_{-i} + \epsilon_i,$$

where p_i is player i 's price and a_i, b_i and c_i are constants and ϵ_i is a random variable.

Our game has the special feature that the quantity of sold products is restricted not only by the demand of products but also by the capacity available. From the assumption that players choose rationally to sell as many products as possible, we know that the quantity of sold products is the smallest of these two. That is

$$q_i = \min\{d_i, \text{capacity available for player } i\}.$$

The main theme in this thesis is to compare some different ways to arrange this capacity available for each player. In all our cases, each player has opportunity to buy own capacity. Whenever a player runs out of its own capacity, the capacity sharing rules determines how much, if any, additional capacity player shall have. We will discuss these rules in the following section.

5.2 Capacity sharing rules

When an operator is investing in capacity, it is actually doing a couple of things. The basic components of capacity are the ones already mentioned. The first component is to buy licence to the frequency bands in auctions, that are currently at least in most countries arranged by government. The second component is to build and maintain the technology, e.g., broadcast towers to make the network work. These two components are dependent on each others since particular frequency bands have special features and they require different kind of technology to receive the full benefit of those features. Modeling these is a totally another issue and we assume that operators can handle this without remarkable losses of capacity. As already mentioned, we represent each operators proprietary capacity as just one decicion parameter in this thesis.

Another thing that might affect a difference between produced capacity and capacity available for customers is that the cognitive radio technology needs itself some source of information about available frequencies. This might be solved in a couple of ways. First, a device which needs a connection could be sensing different frequencies until it finds a free one. This solution has a problem that whenever there are lots of devices sensing the frequencies, it causes interference and thus lowers the efficiency of those frequencies. Buying sensing technology to each device might also be expensive to the customers. Another option is that there is a database which has the information of which frequencies are being used at the moment. Whenever a device needs a free frequency, it should contact the database and receive the information from there.

This also causes some losses to the network. However, neither of these losses are taken into account in this thesis.

Anyway, the new cognitive radio technology allows devices to use frequencies which are unused at a moment. Since there is still lack of free frequencies, we need some rules to divide the unused capacity. We investigate three different rules of sharing capacity whenever a player runs out of its own capacity. These rules are *No sharing* denoted by NS, *whitespace* denoted by WS and *secondary use* denoted by SU. These rules are described in detail in their own sections below.

5.2.1 No sharing (NS)

The first suggestion is the current state. Each player is only allowed to serve their customer only with their own frequencies. If the demand becomes higher than how much capacity the player has bought, the quantity of products sold is exactly the amount they have bought capacity. So the relation between quantity, demand and capacity is

$$q_i = \min\{d_i, k_i\}$$

for each player i.

5.2.2 Whitespace (WS)

Another suggestion is that there is a bandwidth which is not licensed to any specific player, but it is owned by a third party, for example the government. We call this capacity whitespace. Whenever a player runs out of its own capacity, he can borrow capacity from this whitespace. Obviously, we need rules in case there is more over demand than there is this whitespace capacity available. Our suggestion as a dividing rule is that the whitespace capacity is divided in the ratio of over demands of players. By over demand we mean the amount of demand exceeding the capacity. This dividing rule has the property that whenever a new customer chooses between players with demand greater than their own capacity, he has equal chance of getting served whatever he chooses.

We denote this whitespace capacity by k_{WS}. Whitespace has upkeep costs as well as normal capacity. For simplicity we have decided these costs to be covered with constant taxes from each player, not depending on how much whitespace capacity player uses. So there will not be a cost term in the payoff function since adding constant to a payoff function changes nothing. So given this whitespace rule, the relation between quantity, demand and capacity is

$$S = \sum_{j=1}^{n} (d_j - k_j)^+$$
$$q_i = \min\{d_i, \frac{(d_i - k_i)^+}{S} k_{WS} + k_i\},$$

where $(x)^+$ means $\max(x, 0)$ and S is a notation for the sum of the over demands of players. If k_{WS} is zero, whitespace rules equals the NS rules.

5.2.3 Secondary use (SU)

The last suggestion is that there would not be a bandwidth licensed to a third party. Instead of that players use each other's bandwidths whenever there are idle frequencies detected. This rule has an assumption that players are not allowed to fill their own capacities whenever their own demand does not fill it. Secondary use can be expressed such that there is a bandwidth similar to the whitespace, but the size of it is determined when the price game is played. Again we need some rule how to divide the capacity whenever there is more over demand than there is capacity available to be divided. Here we use the same dividing rule as in whitespace case with exception that capacity being shared is calculated from other players' capacities which are not in use. Thus, the relation between quantity, demands and capacities is

$$S = \sum_{j=1}^{n} (d_j - k_j)^+$$
$$k_{SU} = \sum_{j=1}^{n} (k_j - d_j)^+$$
$$q_i = \min\{d_i, \frac{(d_i - k_i)^+}{S} k_{SU} + k_i\},$$

where S is the same as in WS case and k_{SU} is the sum of capacities free to be used by other players. Here it becomes extremely necessary to have some stochastics in demands since it would be irrational for any firm to buy capacity without using it and hence these rules would reduce to the NS rules.

5.3 Indexes for comparison

As the main theme of this thesis is to choose which of these rules have the most attractive properties, we need to define some way to compare those to each other.

There are three groups involved to this market: the customers, the operators or the players and the government. Customers are interested in prices. However, it is not quite clear how to compare two cases where some players' prices are lower than the others'. So we have decided the sum of demands to represents the happiness of the customers. Players are interested to maximize their profits which are equal to the payoffs. The government is willing to make the network as efficient as possible. That means the ratio between total quantity of sold products and total amount of capacity licenced to this market. For each of these targets we have generated an index. They are calculated with the values in an equilibrium of the game.

$$
\begin{aligned}
I_e &= \frac{\sum q_i}{k_{WS} + \sum k_i} \\
I_d &= \sum d_i \\
I_p &= \sum \pi_i,
\end{aligned}
$$

where index e refers to efficiency, d refers to demand and p refers to profits.

6 Analysis of the model

6.1 No sharing

Let us first assume that we have no stochastics at all. We may assume that in equilibrium point, no one will buy capacity that will be unused in stage 2. That would be dominated by the strategy to buy less capacity and still set the same price. Also the demand will not be higher than capacity since setting price higher and still selling the same quantity of product dominates that strategy. So we actually have for the equilibrium strategies

$$d_i(p^*, k^*) = k_i^* \quad \forall i \in \{1, \dots, n\}. \tag{3}$$

By putting the definition of d_i to the equation system (3), we can write it to a matrix form

$$Mp^* = \beta,$$

where $\beta = (\beta_1, \dots \beta_n)^T$ and for each i

$$\beta_i = \frac{k_i^* - a_i}{c_i}$$

and

$$(M)_{ij} = \begin{cases} -\frac{b_i}{c_i}, & \text{when } i = j \\ 1, & \text{when } i \neq j. \end{cases}$$

Thus, we get a representation for the equilibrium price profile

$$p^* = M^{-1}\beta. \tag{4}$$

Now the equilibrium point must satisfy the first order optimality condition

$$\frac{\partial \pi_i}{\partial k_i} = 0. \tag{5}$$

Let us assume that equation (3) holds actually true whenever k_i is in some neighborhood of k_i^*. Then we can apply the above result to (5) and receive linear equation system which gives $k_1^*, \ldots, k_n^*$ and putting those to equation (4) gives $p_1^*, \ldots, p_n^*$.

Thus, we are able to solve this case analytically. Unfortunately we are not as lucky in the other cases.

6.2 Price game with whitespace capacity

In this case there is common capacity k_{WS}. The rules of this game are represented in the following set of equations

$$
\begin{aligned}
d_i &= a_i - b_i p_i + c_i \sum p_{-i} \\
q_i &= \begin{cases} \dfrac{d_i - k_i}{S} k_{WS} + k_i, & \text{when } d_i \geq k_i \text{ and } S \geq k_{WS} \\ d_i, & \text{otherwise} \end{cases} \\
S &= \sum_{j=1}^{n} (d_j - k_j)^+ \\
\pi_i &= p_i q_i - g k_i^2.
\end{aligned}
$$

We split the analysis of the price game into two parts according to the amount of whitespace capacity provided. First, we analyze the case where there is enought common capacity to cover the demands without any firm buying proprietary capacity. From that, we get the demands for unconstrained price game. So we get the threshold value whether the common capacity is being totally used or not.

6.2.1 Large common capacity

Let us first assume that k_{WS} is large. Problem reduces to unconstrained problem where buying capacity is dominated by not buying, so no player will buy own capacity. By using the first order optimality condition we get the equilibrium by solution of the following system

$$
\left. \frac{\partial \pi_i}{\partial p_i} \right|_{(p_1, \ldots, p_n) = (p_1^*, \ldots, p_n^*)} = a_i - 2b_i p_i^* + c_i \sum p_{-i}^* = 0 \tag{6}
$$

which can be represented as matrix equation

$$M'p^* = \beta',$$

where M' and β' depend on the parameters of the model. Next, we calculate the total quantity of sold products in this game. We get

$$\begin{aligned}\sum_{j=1}^{n} q_j(p^*) &= \sum_{j=i}^{n} b_j p_j^* \\ &= b^T p^* \\ &= b^T M^{-1}\beta,\end{aligned}$$

where in the first equation we used that $q_i(p^*) = d_i(p^*)$ according to our assumption of large whitespace and $d_i(p^*) = b_i p_i^*$ given from equation (6).

So if the whitespace capacity satisfies condition $k_{WS} \geq b^T M'^{-1}\beta'$ we have enough capacity that no-one have to buy it. On the other hand if we have less than $b^T M'^{-1}\beta'$ common capacity, it will be totally used in the equilibrium.

6.2.2 Small common capacity

If we have less than $b^T M'^{-1}\beta'$ common capacity then we must have $d_i(p^*) \geq k_i$ and $S \geq k_{WS}$ according to reasons mentioned in the last section. So by writing the first order optimality condition we get

$$\frac{\partial \pi_i}{\partial p_i}\bigg|_{(p_1,\ldots,p_n)=(p_1^*,\ldots,p_n^*)} = q_i(p^*) + p_i^*\Big(k_{WS}\frac{-b_i S - (d_i(p^*) - k_i)(\sum c_{-i} - b_i)}{S^2}\Big) = 0,$$

where we have n equations of degree 2 and n unknowns. This is easy to solve numerically but not analytically whenever $n \geq 3$. Also we know that 2nd order equations produce always 2 answers.

Next we examine these 2 answers. Define function $f(p_i)$ by putting q_i by $\frac{d_i - k_i}{S} k_{WS} + k_i$ in π_i. Now f has a singularity at $S = 0$ and is differentiable everywhere else. Let us denote this value by p_i'. Now $\lim_{p_i \to p_i'^-} f(p_i) = -\infty$ and $\lim_{p_i \to p_i'^+} f(p_i) = \infty$. Also $\lim_{p_i \to -\infty} f(p_i) = -\infty$ and $\lim_{p_i \to \infty} f(p_i) = \infty$. So we know that there must be an extremal value in both sides of the singularity. We know that there is only 2 points where derivative is 0 so there is exactly one on each side. So we always must choose the lower of those 2 answers to receive the correct one (the other one has the property $S < 0$). So we have always a unique best response for given actions of other players.

So, we end up using numerical methods already in whitespace case without stochastics. The secondary use case is just more complex than whitespace case and adding stochastics makes equations even more complex to handle analytically so this clearly is a numerical problem.

6.3 Full game with WS and SU rules

Our game is a two-stage game. At the second stage we know how much capacity each player has and hence it is possible to calculate each player's payoff for every outcome of the price game. If we could solve the price game analytically, we could just replace the price in the payoff function by the solution of the price game and the problem would be reduced to a single stage game. However, there is no analytical solution to the price game and hence prices are some unknown function of capacities. In the next section, we introduce some numerical methods to solve this problem.

7 Algorithm

In this chapter we construct an algorithm to solve our two-stage game numerically. First, we need to be able to solve subgames, i.e., the price games whenever we have fixed capacities. Second, given these results we solve the capacity game. However, there are M^n profiles of capacities, where M is the amount of possible capacity values for each player. Since we want the discretization to be relatively dence near equilibrium, M is a large number. Also, the algorithm does not use solutions of subgames with all profiles of capacity. To avoid computing subgames which are not necessary, we solve price games only whenever the solution is needed by the algorithm that is solving the capacity game.

In the following sections we introduce two algorithms, one that was introduced by Basar and another introduced by Porter, Nudelman and Shoham. They are applied to our game in the following way:

1. Capacity game
   Basar's algorithm
2. Price game
   Basar's algorithm until cycle detected
   Porter–Nudelman–Shoham algorithm

7.1 Main algorithm

Our main algorithm is introduced by Basar in [6]. It is based on best response dynamics. Best response function is defined in the end of chapter 3. We denote the function which returns each players' best response for a given profile of strategies $s = (s_1, \dots, s_n)$ by

$$BR(s) = (BR_1(s_{-1}), \dots, BR_n(s_{-n})).$$

In numerical calculations, we define the best response functions to be single valued. Thus, if there are more than one maximizing strategy, the best response function simply chooses one.

Whenever there is a pure strategy Nash equilibrium, it is a fixed point of the BR function. To solve fixed point of a function we can use simple algorithm where function itself gives the next iteration step. In game theoretic context this is called Cournot adjustment [3] and can be represented by

$$s^{k+1} = BR(s^k). \tag{7}$$

However, this algorithm has pure convergence properties and it too often ends up oscillating. Algorithm introduced by Basar has an additional property that it does not take only the best response into consideration, but also the earlier step of iteration in ratio of a relaxation parameter $\alpha \in [0, 1)$. The iteration step in this algorithm is

$$s^{k+1} = \alpha s^k + (1 - \alpha)BR(s^k). \tag{8}$$

This algorithm gives much faster convergence since the effect of oscillation is smaller. Basar has studied convergence properties of this algorithm in [6]. Under certain conditions it can be proven that this algorithm converges and the convergence speed is much faster than Cournot adjustment (7). Unfortunately the payoff function in our model does not satisfy the conditions that Basar has been analyzing. Actually, there are situations in this game such that there is no Nash equilibrium with pure strategies. Anyway, in practice (8) converges much faster than (7) and thus we use it.

The benefits of this relaxation term are mostly received near an equilibrium point where the oscillation happens. Thus, we actually change the value of α to start near 0 and increase as a function of ordinal number of iteration step.

7.2 Problems

In some cases, even the Basar's algorithm ends up oscillating along some cycle. This is a good indicator that there might not be pure strategy equilibrium at all. In general, mixed strategy Nash equilibrium in a game with continuous strategies is following:

A profile of probability distributions over sets of actions forms a Nash equilibrium if no player benefits by deviating alone from his strategy whenever other players keep playing their strategies.

To check that the condition holds, we actually need to compare given strategies only to all pure strategies. The reason is that expected payoff of a mixed strategy is a convex combination of payoffs of pure strategies and thus it is never strictly higher than the payoffs of all pure strategies.

So we have a method to check whether or not a given strategy is a Nash equilibrium. However, this is not really giving much advice how to find it.

Next we need candidates to be tested. As a result of Basar algorithm iteration, we end up in a situation, where the best response strategies of each player oscillate between some lower and upper bounds. So we might expect there to be an equilibrium in that polyhedral set containing at least one mixed strategy.

According to this assumption we create a following discretized game. Let $\overline{s_i}$ be the upper bound and $\underline{s_i}$ be the lower bound of player i 's strategy. Choose a profile of amounts of points in discretizations $(m_1, \ldots, m_n) \in \mathbb{N}^n$ and create new action sets $(\mathbb{S}'_1, \ldots, \mathbb{S}'_n)$ such that

$$\mathbb{S}'_i = \{\underline{s_i} + \frac{\overline{s_i} - \underline{s_i}}{m_i - 1} k \mid k = 0, \ldots, m_i - 1\}$$

for each player i. The payoff functions stay the same. So we have a discrete and finite game and next task is to find an equilibrium of that game.

7.3 Solving normal-form games

The second algorithm is designed to solve any one-stage n player finite normal-form game. It is introduced by Porter et.al. in [8].

This algorithm is basically a brute force algorithm with some improvements. It goes through all profiles of *supports* one by one until it finds a Nash equilibrium. Support specifies the set of actions for a player played with positive probability. The improvements are that the algorithms tests whether certain conditions, which prevent certain support to be in an equilibrium, hold. This actually lowers dramatically the amount of cases to go through. The structure of the algorithm is following.

While a Nash equilibrium is not found, use *recursive-backtracking* for every profile $(x_1, x_2, \ldots, x_n) \in \mathbb{N}^n$, where x_i is size of support for player i. Naturally the size of each x_i is bounded above by the size of set $\mathbb{S}'_i$.

Recursive-backtracking takes a profile of *domains* $D = (D_1, D_2, \ldots, D_n)$, where each domain D_i is a set of supports for player i. Algorithm takes also an index which tells which player's domain are we consentrating at the moment. Always when we call this function with a new profile x we use index 1 and for each i, D_i is a set of all supports, or subsets of $\mathbb{S}'_i$, containing exactly x_i elements.

When recursive backtracking is called with profile of domains D and index i, the idea is for each support $S_i \in D_i$ to call the function itself with index $i+1$ and profile D^* which is the same as D except that D_i is replaced by $\{S_i\}$. Thus, it follows that for every $j < i$ domain D_j contains only one support.

Whenever the algorithm ends up in a situation where each D_i contains one support (which happens when the function is called with index $n + 1$) we have for each player the set of actions with positive probability in an equilibrium and thus a relatively easy constraint satisfaction problem. If it has a solution, that is a Nash equilibrium and the algorithm terminates. Otherwise it continues with next branch of the algorithm.

Actually, using just the part of the algorithm explained thus far would find a Nash equilibrium. However, there is also another part which reduces the amount of calculations. This part is called *Iterated removal of strictly dominated strategies* (IRSDS). To introduce it we define the *conditionally dominated actions*.

Definition: Conditionally dominated actions. An action $a_i \in \mathbb{S}'_i$ is conditionally dominated, given a profile of sets of available actions $R_{-i} \subset \mathbb{S}'_{-i}$ for the remaining players, if the following condition holds:

$$\exists a'_i \in \mathbb{S}'_i \quad \forall a_{-i} \in R_{-i} : u_i(a_i, a_{-i}) < u_i(a'_i, a_{-i}).$$

IRSDS algorithm takes as input the profile of domains D. It removes iteratively from each D_i supports which contains conditionally dominated actions given that the set of available actions is actions in supports still in D_{-i}. Removing continues until some D_i becomes empty or there are no conditionally dominated actions left in any support of any D_i. IRSDS algorithm is called in the beginning of the recursive-backtracking algorithm to be applied every time when we fix D_i to be some particular support S_i. In IRSDS, if some D_i becomes empty, it is impossible that there would be a profile of supports to form a Nash equilibrium in the original D. Thus we can terminate the whole branch in the recursive-backtracking algorithm.

Algorithm to solve normal-form games

for all $x = (x_1, \ldots, x_n)$, sorted in increasing order of primarily by $\sum_i x_i$ and secondarily by $\max_{i,j}(x_i - x_j)$ **do**
  $\forall i : S_i \leftarrow NULL$ //uninstantiated supports
  $\forall i : D_i \leftarrow \{S_i \subseteq \mathbb{S}'_i : \lvert S_i \rvert = x_i\}$ //domain of supports
  if Recursive-Backtracking$(S, D, 1)$ returns a NE p **then**
    Return p

Procedure 1: Recursive-Backtracking

Input: $S = (S_1, \ldots, S_n)$: a profile of supports
  $D = (D_1, \ldots, D_n)$: a profile of domains
  i: index of next support to instantiate
Output: A Nash equilibrium p, or *failure*
if $i = n + 1$ **then**
  if Feasibility Program is satisfiable for S **then**
    Return the found NE p
  else
    Return *failure*
else
  for all $d_i \in D_i$ **do**
    $S_i \leftarrow d_i$
    $D_i \leftarrow D_i - \{d_i\}$
    if $IRSDS((\{S_1\}, \ldots, \{S_i\}, D_{i+1}, \ldots, D_n))$ succeeds **then**
      if Recursive-Backtracking$(S, D, i + 1)$ returns NE p **then**
        Return p
  Return *failure*

Procedure 2: Iterated Removal of Srictly Dominated Strategies (IRSDS)

Input: $D = (D_1, \ldots, D_n)$: profile of domains
Output: Updated domains, or *failure*
repeat
  $changed \leftarrow false$
  for all $i \in N$ **do**
    for all $a_i \in \cup_{d_i \in D_i} d_i$ **do**
      for all $a_i' \in \mathbb{S}_i'$ **do**
        if a_i is conditionally dominated by a_i', given $\cup_{d_{-i} \in D_{-i}} d_{-i}$ **then**
          $D_i \leftarrow D_i - \{d_i \in D_i : a_i \in d_i\}$
          $changed \leftarrow true$
          if $D_i = \emptyset$ **then**
            Return *failure*
untill $changed = false$
Return D

Feasibility Program
Input: $S = (S_1, \ldots, S_n)$, a support profile **Output**: NE p, if there exists both a strategy profile $p = (p_1, \ldots, p_n)$ and a value profile $v = (v_1, \ldots, v_n)$ such that: $\forall i \in N, a_i \in S_i : \sum_{a_{-i} \in S_{-i}} p(a_{-i}) u_i(a_i, a_{-i}) = v_i$ $\forall i \in N, a_i \notin S_i : \sum_{a_{-i} \in S_{-i}} p(a_{-i}) u_i(a_i, a_{-i}) \leq v_i$ $\forall i \in N : \sum_{a_i \in S_i} p_i(a_i) = 1$ $\forall i \in N, a_i \in S_i : p_i(a_i) \geq 0$ $\forall i \in N, a_i \notin S_i : p_i(a_i) = 0$

Porter et.al. state in their article [8] that this algorithm has the properties of being sound and complete. This means that every solution this algorithm returns is actually a solution to the given problem and whenever a solution exists, this algorithm will eventually find it. The third property mentioned by them is low computational complexity which this algorithm does not satisfy. Of course the other two criteria are also dependent of whether the constraint satisfaction algorithm has those same properties.

7.4 Improving algorithms efficiency

So now we have an algorithm to solve the price game for any outcome of the capacity game. Next task is to solve the capacity game. Again, we use Basar's algorithm [6]. The amount on calculations in the algorithm we are using is huge. We make some improvements to the algorithm to make it quicker. In the main algorithm we are optimizing due to one parameter only at a time and we use two ways to do it:

1) Using the golden section algorithm.

2) Computing the whole range with constant step size and picking the highest.

The amount of function evaluations in the golden section method is remarkably less than in the second method. Since each function evaluation means solving a subgame numerically, it requires lots of calculations. However, the only method to get the global maximum is the second method since the function to be maximized does not satisfy conditions to provide unique maximum.

In this thesis, we use the golden section algorithm in early stages of iterations to get quickly near an equilibrium. Whenever an equilibrium is assumed to be near, we start using the second mathod to be as accurate as possible.

Another essential thing is determining the amount of strategies in normal-form game for each player. In their article [8] Porter et.al. states that this problem is in complexity class TFNP which is developed by N. Megiddo and C. Papadimitriou in [21]. TFNP stands for "Total Function Nondeterministic Polynomial". For similar reasons to ones mentioned above we have limited the use of normal-form game to only those cases when we are close to the equilibrium. We have defined the amount of strategies used such that the smaller the difference between lower and upper bound is, the less options there will be for that player. No more than six options will be available for any player.

8 Numerical Results

As predicted in chapter 7, Basar's algorithm ends up oscillating between some values. Whenever a price game has a Nash equilibrium $p^* = (p_1^*, \dots, p_n^*)$ with pure strategies, it must satisfy the condition

$$p_i^* \in BR_i(p_i^*) \quad \forall i \in \{1, \dots, n\}.$$

In two player price game, we can figure out all pure strategy Nash equilibria by drawing the best response functions $BR_1(p_2)$ and $BR_2(p_1)$ in the same (p_1, p_2)-coordinate system. Pure strategy equilibria are intersection points of those functions, so the intersections of the curves are exactly the Nash equilibria of the game. An example of best response functions drawn in the same coordinate system, and not intersecting, is represented in Figure 7.

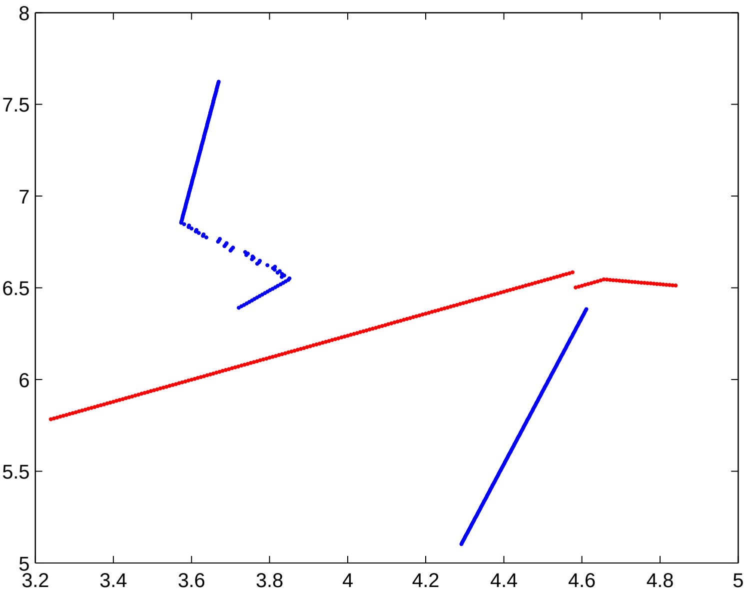

Figure 7: Best response functions in price game not intersecting. Rules implied are SU-rules and parameter values used are $a_1 = 16$, $a_2 = 8$, $b_1 = 2$, $b_2 = 1.5$, $c_1 = 1.2$, $c_2 = .375$, random variable for player i used is from discretized uniform distribution with four points and values from zero to ϵ_i, where $\epsilon_1 = \epsilon_2 = 0.1$. As this is just a price game, the fixed capacity values are $k_1 = 8.5225$ and $k_2 = 3.6768$.

As seen in figure 7 best response functions do not intersect and thus there is no Nash equilibrium with pure strategies in that game. This kind of behavior is typical in the price game with two players. When there are more players, the functions BR_i have $n-1$ parameters and thus are hypersurfaces in n dimensional space. Thus, they are too complex to be represented as simply as in the two player case. Anyway, price games with more players have same kind of behavior as the two player case: algorithm keeps oscillating through paths. Also since it is typical that there exists no pure strategy equilibrium in two player game, it should be typical in many player game as well.

Sometimes there are lots of Nash equlibria in a price game with two players. Picture of best response functions intersecting more than once is represented in Figure 8. In these games the algorithm simply chooses the one Nash equilibrium it hits first. The algorithm is not interested if there are more equilibria.

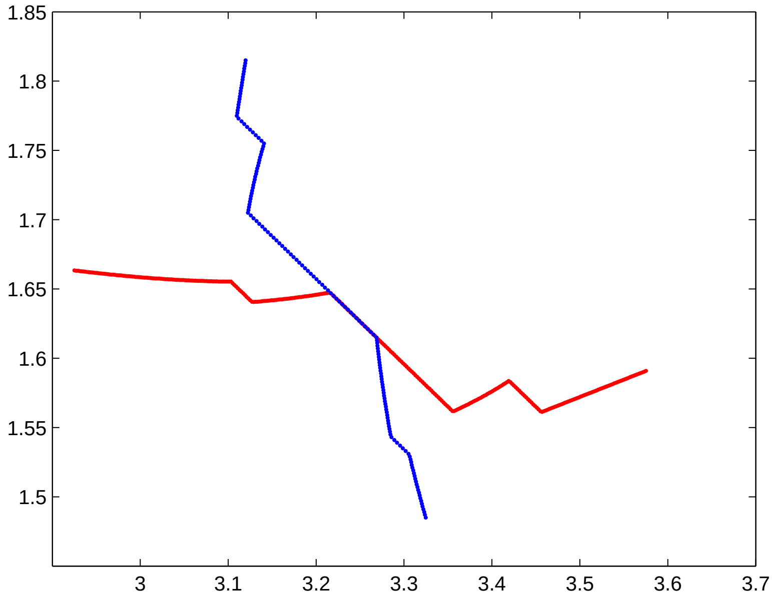

Figure 8: Best response functions in price game intersecting more than once. Rules implied are SU-rules and parameter values used are $a_1 = 4$, $a_2 = 8$, $b_1 = 2$, $b_2 = 1.5$, $c_1 = 0.5$, $c_2 = .375$, random variable for player i used is from discretized uniform distribution with four points and values from zero to ϵ_i, where $\epsilon_1 = \epsilon_2 = 0.1$. As this is just a price game, the fixed capacity values are $k_1 = 2.106$ and $k_2 = 3.501$.

8.1 Two player game

The used parameter values are represented in Table 1. The random variable is discrete uniform distribution between zero and ϵ_i. We use the same ϵ_i for each player. The payoff is computed as an expected value due to this parameter. The cost function of capacity is $\frac{1}{50}k_i^2$. The parameter values have been chosen such that the elasticities in the equilibrium are reasonable to telecommunication business.

Table 1: Parameter values used in two player game.

a_1	2.0	4.0	8.0	16.0
a_2	8.0			
b_1	1.5	2.0		
b_2	1.5			
c_1	0.25	0.6		
c_2	0.5			
ϵ_i	0.1			
k_{WS}	0.2	0.4	0.6	

Results of these calculations are represented in Tables 2 - 5. Each cell includes the efficiency index value on the top, profit index value in the middle and demand index value on the bottom. The biggest value in each row is highligted such that the biggest efficiency is underlined, the biggest profit index is bolded and the biggest demands are framed. Many cases were terminated by reaching the maximum amount of iterations in capacity game. Those which ended before maximum iterations are marked with $*$.

Table 2: Index values with two players with low b_1 and low c_1.

	NS	*SU*	*WS small*	*WS medium*	*WS large*
$a_1 = 2$	0.8921 14.2765 5.0365	<u>1.0000</u> 12.8834 4.7959	0.9144 14.3134 5.1779	0.9164 * 14.3838 5.2386	0.9710 **14.4171** $\boxed{5.3351}$
$a_1 = 4$	0.8735 18.7205 5.9779	<u>1.0000</u> 18.0484 5.5458	0.8999 * 18.7956 6.1681	0.9414 18.9801 6.2172	0.9442 **19.0281** $\boxed{6.2976}$
$a_1 = 8$	0.8496 * 32.0374 7.8711	<u>0.9861</u> **32.8239** 7.0774	0.8731 32.2625 8.0920	0.8951 32.4767 8.1220	0.9327 32.5858 $\boxed{8.2824}$
$a_1 = 16$	0.8305 76.5347 11.6671	0.8876 76.4808 10.0979	0.8465 76.9443 11.8579	0.8762 77.3785 11.9984	<u>0.8881</u> **77.5836** $\boxed{12.1322}$

Table 3: Index values with two players with high b_1 and low c_1.

	NS	*SU*	*WS small*	*WS medium*	*WS large*
$a_1 = 2$	0.9046 * 13.8062 5.1943	1.0000 12.0815 4.8360	0.9210 13.8609 5.3178	0.9791 13.9157 $\boxed{5.4156}$	0.9628 * **14.0127** 5.3560
$a_1 = 4$	0.8924 * 17.3324 6.1258	1.0000 16.1860 5.4523	0.9103 17.4138 6.2757	0.9549 **17.5238** 6.3586	0.9647 17.4953 $\boxed{6.5013}$
$a_1 = 8$	0.8767 27.6831 7.9949	1.0000 27.8707 6.7494	0.8967 27.8277 8.2109	0.9288 27.9709 8.3030	0.9528 **28.0085** $\boxed{8.4663}$
$a_1 = 16$	0.8626 * 61.6331 11.7774	0.9163 60.1238 9.3835	0.8816 61.9662 11.8563	0.8879 62.1603 12.0323	0.9107 **62.3784** $\boxed{12.0751}$

Table 4: Index values with two players with low b_1 and high c_1.

	NS	*SU*	*WS small*	*WS medium*	*WS large*
$a_1 = 2$	0.9383 **19.4758** 5.7809	1.0000 18.2824 5.5295	0.9561 19.3868 5.9371	0.9754 19.3907 5.9926	0.9867 * 19.1809 $\boxed{6.1580}$
$a_1 = 4$	0.9105 * 26.5594 6.6783	0.9728 **26.9407** 6.4236	0.9307 26.4478 6.8640	0.9603 26.2047 7.0822	0.9589 26.1861 $\boxed{7.1248}$
$a_1 = 8$	0.8797 46.2533 8.5089	0.9630 **46.4029** 7.8119	0.8951 46.1371 8.6996	0.9133 46.0565 8.8155	0.9235 * 45.7044 $\boxed{9.0696}$
$a_1 = 16$	0.8447 **107.8643** 12.1423	0.9950 106.3940 12.5648	0.8609 * 107.6232 12.4479	0.8707 107.4368 12.4830	0.8876 106.8120 $\boxed{12.9577}$

Table 5: Index values with two players with high b_1 and high c_1.

	NS	*SU*	*WS small*	*WS medium*	*WS large*
$a_1 = 2$	0.9415 * 19.5521 5.9288	<u>1.0000</u> **19.6374** 5.7298	0.9843 * 19.2985 6.2397	0.9688 19.1210 6.3677	0.9802 19.0460 $\boxed{6.4232}$
$a_1 = 4$	0.9300 * 25.2035 7.0902	<u>1.0000</u> * **25.2228** 6.2632	0.9460 25.0493 7.2121	0.9566 24.8018 7.3863	0.9753 24.6225 $\boxed{7.4975}$
$a_1 = 8$	0.9075 **41.1231** 8.8871	<u>1.0000</u> * 36.7729 7.7310	0.9316 40.8699 9.0296	0.9322 40.5257 9.1748	0.9508 40.1425 $\boxed{9.3889}$
$a_1 = 16$	0.8799 **89.5235** 12.4830	<u>1.0000</u> 75.6532 11.7360	0.8987 88.9640 12.6895	0.9112 88.0679 13.0288	0.9147 87.3616 $\boxed{13.0767}$

We can see from the results that efficiency is highest, nearly 1 in secondary use rules. The only exception to this is in Table 2 last row where efficiency is about the same as in large whitespace case. Between whitespace cases efficiency is increasing in terms of whitespace capacity available. To this there are three exceptions: in Table 3 first row, in Table 4 second row medium case is too high or large case is too low, and in Table 5 first row the small whitespace case has too high efficiency.

Demands are also increasing in terms of whitespace capacity. The only exception is in Table 3 second row. In secondary use case, demands are lower. There are also one exception to this in Table 4 last row.

The players profits are nearly the same in each set of rules in two player game.

8.2 Three player game

Values used in three player case is given in Table 6. The random variable and the cost function of capacity are the same as in two player case.

Table 6: Parameter values used in three player game.

a_1	2.0	4.0	8.0	16.0
a_2	8.0			
a_3	8.0			
b_1	1.5	2.0		
b_2	1.5			
b_3	1.5			
c_1	0.25	0.6		
c_2	0.5			
c_3	0.5			
ϵ_i	0.1			
k_{WS}	0.2	0.4	0.6	

The results of calculations in three player case is given in Tables 7 - 10. Cells includes the same indexes as in two player game. Also the notation is the same as in two player game.

Table 7: Index values with three players with low b_1 and low c_1.

	NS	*SU*	*WS small*	*WS medium*	*WS large*
	0.8926	$\underline{1.0000}$	0.8969	0.9199	0.9450
$a_1 = 2$	32.8912	21.8348	32.6576	33.0048	**33.1052**
	9.5635	6.8095	$\boxed{9.9917}$	9.6727	9.8569
	0.8913	$\underline{1.0000}$	0.8876	0.8116	0.9377
$a_1 = 4$	* 37.8553	29.2477	37.4515	37.1990	**38.1976**
	10.6691	6.3460	$\boxed{11.1858}$	11.1010	10.9208
	0.8908	$\underline{1.0000}$	0.9168	0.8902	0.7050
$a_1 = 8$	52.0880	38.4585	**52.4540**	51.2710	50.1902
	12.8518	8.5176	13.0455	13.4028	$\boxed{13.6024}$
	0.8911	$\underline{1.0000}$	0.9153	0.9069	0.9204
$a_1 = 16$	97.5902	74.6467	**98.3779**	97.4363	98.3191
	17.2623	11.3059	17.2926	$\boxed{17.5658}$	17.4864

Table 8: Index values with three players with high b_1 and low c_1.

	NS	*SU*	*WS small*	*WS medium*	*WS large*
	0.8975	$\underline{1.0000}$	0.9312	0.9147	0.9450
$a_1 = 2$	32.3251	21.1754	32.4598	32.3586	**32.5407**
	9.7221	7.8493	9.9454	9.8890	$\boxed{9.9584}$
	0.8983	$\underline{1.0000}$	0.9347	0.9204	0.9228
$a_1 = 4$	36.2625	27.8029	**36.5508**	36.3796	35.9565
	10.7770	7.5355	10.8981	10.8892	$\boxed{11.4017}$
	0.8993	0.9369	0.8396	0.9185	$\underline{0.9404}$
$a_1 = 8$	* 47.3031	43.3874	46.3259	**47.4290**	47.1599
	12.8709	7.9650	$\boxed{13.6249}$	13.0345	13.4772
	0.9024	$\underline{1.0000}$	0.9237	0.9158	0.9278
$a_1 = 16$	* 81.9935	46.5184	82.4888	82.1334	**82.5473**
	17.0978	12.7223	17.2033	17.2548	$\boxed{17.2924}$

Table 9: Index values with three players with low b_1 and high c_1.

	NS	*SU*	*WS small*	*WS medium*	*WS large*
	0.9009	$\underline{1.0000}$	0.9354	0.9260	0.9497
$a_1 = 2$	* 39.1094	31.9138	39.1993	**39.2218**	* 39.1976
	10.4414	8.0960	10.6069	10.5201	$\boxed{10.6741}$
	0.9045	$\underline{1.0000}$	0.9294	0.9246	0.9377
$a_1 = 4$	* 46.2498	34.5259	**46.3702**	46.2503	46.0554
	11.5092	9.5146	11.6678	11.6548	$\boxed{11.8877}$
	0.9016	$\underline{1.0000}$	0.9033	0.9127	0.9373
$a_1 = 8$	65.2247	61.6239	**65.4984**	64.6713	62.3783
	13.6855	10.1456	13.7462	13.9956	$\boxed{14.9467}$
	0.8953	$\underline{1.0000}$	0.9074	0.9074	0.9209
$a_1 = 16$	122.2191	97.7001	**122.6544**	122.2752	122.4300
	17.9540	13.2709	18.0023	18.0785	$\boxed{18.1682}$

Table 10: Index values with three players with high b_1 and high c_1.

	NS	*SU*	*WS small*	*WS medium*	*WS large*
	0.9090	$\underline{1.0000}$	0.9438	0.9324	0.9465
$a_1 = 2$	* 39.4991	28.3780	39.4090	**39.5062**	39.0980
	10.4421	8.1868	10.7775	10.6067	$\boxed{11.0230}$
	0.9122	$\underline{1.0000}$	0.9372	0.9230	0.9550
$a_1 = 4$	* 44.9724	32.1378	45.0327	**45.1686**	44.7461
	11.9095	9.3473	12.0247	11.8817	$\boxed{12.2655}$
	0.9136	$\underline{1.0000}$	0.9350	0.9304	0.9433
$a_1 = 8$	**60.3708**	53.7587	60.2314	60.0899	60.1759
	13.9806	9.1591	14.1762	14.1423	$\boxed{14.2289}$
	0.9097	$\underline{1.0000}$	0.9272	0.9254	0.9367
$a_1 = 16$	**105.1372**	72.4515	104.9083	104.5868	104.6351
	18.0754	11.3637	18.2489	18.3102	$\boxed{18.3991}$

From the results we get that efficiency is highest with secondary use rules. The only exception to this is in Table 8 third row, where it is slightly lower than with large whitespace case. Also comparing no sharing to any whitespace cases, the efficiency is larger with whitespace. There are two exceptions to this, in Table 7 and Table 8 on third row. However, between whitespace cases we cannot see a clear correspondence to efficiency.

Demands are lowest in secondary use cases with no exceptions. Also no sharing has lower demand than any whitespace case. The only exception is Table 10 second row. Between whitespace cases there are no clear correspondence to demands.

Profits are lower in secondary use cases. The gap is quite small when we have a_1 value 8.0, but wider in other cases. Between other cases there are no clear differences in profits.

9 Discussion

As many of the cases analyzed are not terminated before the maximum amount of iterations, we could consider using mixed strategies also in the capacity game. However, this would be unrealistic since frequencies are licenced in an auction and thus players are forced to reveal their capacity buying strategy step by step while others are still able to buy it.

In game theory we must assume players to be rational. In this kind of two-stage game where we need to use mixed strategies in second stage and possibly also in the first stage, it seems quite unrealistic to assume that other operators also goes through analysis represented in this thesis especially when we do not really know the exact demand function parameters. Near equilibrium payoff functions are also really flat, so finding equilibrium is slowly process and benefits are really low when comparing values near the equilibrium. Also if other operators do not play equilibrium strategy, there are better strategies for other operators instead of playing the equilibrium strategy.

Our choice to handle the market model as a static game could also be replaced with dynamic game like many other researchers have done, for example [10] and [11]. This would be quite natural since in telecommunication market there are always history of actions of other operators. So instead of solving equilibrium operators only have to figure out how to respond to last actions of others.

10 Conclusions

In this thesis we computed the Nash equilibrium of telecommunication market model with three different possible rules implied. We examined the game analytically, but ended up problems with too complex equation systems. We constructed an algorithm which uses a couple of algorithms to perform different tasks. To solve the capacity game we used Basar's algorithm with brute force line search method and also the golden section method to lower the complexity of calculations a bit. To do this, we needed to solve the price game as a subgame in each capacity game point that the algorithm needed. In price games we again used the same algorithm. Whenever we ended up oscillating in the price game we formed a normal-form game which we solved using Porter et.al. algorithm.

For two player games the algorithm was quite useful in terms of execution time. In three player case using "No sharing" or "Whitespace" rules, the algorithm took less than a day to solve a game. With "Secondary use" rules the algorithm took from a couple of days to a few weeks to solve the game. The used computer has one Intel Core 2 Duo E8400 processor, 4GiB of 800MHz DDR2 memory and no discrete graphics card.

As a result we can conclude that secondary use rules implied the highest efficiency while no sharing implied to lowest efficiency. Demands were highest with whitespace rules and lowest with secondary use rules. Profits did not change so much between whitespace and no sharing rules. In three players' secondary use rules profits were remarkably lower than with other rules. So, the government should be willing to change from current state to both secondary use and whitespace. Changing to the secondary use brings lower profits to firms and lower demands, so they will oppose that change. On the other hand, changing to white space should not be resisted by any party since that is the most preferred case to customers and about equally good as current state to firms profits as long as the taxes from whitespace are small enought.

11 References

References

[1] R. Gibbons. A primer in game theory. FT Prentice Hall. 1992.

[2] D. Fudenberg, D. Levine. The theory of learning in games. Second printing. The MIT press. 1999.

[3] D. Fudenberg, J. Tirole. Game theory. The MIT press. 1991.

[4] X. Vives. Oligopoly pricing old ideas and new tools. The MIT press. 2001.

[5] J. Nash. Equilibrium points in n-person games. Proceedings of the National Academy of Sciences 36(1). Pages 48-49. 1950.

[6] T. Basar. Relaxation techniques and asynchronous algorithms for on-line computation of non-cooperative equilibria. Journal of Economic Dynamics and Control 11 (1987). North-Holland. Pages 531-549. 1987.

[7] J. Krawczyk, S. Uryasev. Relaxation algorithms to find Nash equilibria with economic applications. Environmental modelling and assessment 5. Pages 63-73. 2000.

[8] R. Porter, E. Nudelman, Y. Shoham. Simple Search Methods For Finding A Nash Equilibrium. Games and Economic Behavior 63. Pages 642–662. 2005.

[9] A. Klemettilä. Three player oligopoly model to evaluate the economic impact of cognitive radio. Masters thesesis in Aalto University. 2010.

[10] J. Jia, Q. Zhang. Compeptitions and Dynamics of Duopoly Wireless Service Providers in Dynamic Spectrum Market. MobiHoc '08 May 26-30. Pages 313-322. 2008.

[11] Y. Xu, J. Lui, D. Chiu. On Oligopoly Spectrum Allocation Game in Cognitive Radio Networks wiht Capacity Constraints. Computer Networks vol. 54, no. 6. 2010.

[12] A. Mas-Colell, M. Whinston, J. Green. Microeconomic theory. Oxford University press. 1995.

[13] S. Haykin. Cognitive radio: brain-empowered wireless communications. IEEE journal on selected areas in communications, VOL. 23, NO. 2. 2005.

[14] J. Mitola. Cognitive radio: making software radios more personal. IEEE Personal Communivations, VOL. 6. NO. 4. 1999.

[15] B. Fette. Cognitive radio Technology. 2ND edtiton. Elsevier. 2009.

[16] L. Barlemann, S. Mangold. Cognitive radio and dynamic spectrum access. A John Wiley & Sons Ltd. 2009.

[17] D. Cabric, S. Mishira, R. Brodersen. Implementation Issues in Spectrum Sensing for Cognitive Radios. IEEE. 2004.

[18] V. Valenta, R. Maršálek. Survey on spectrum utilization in Europe: measurements, analyses and observations. National ICST conference on cognitive radio oriented wireless Networks and communications. 2010.

[19] T. Weiss, F. Jondral. Spectrum pooling: an innovative strategy for the enhancement of spectrum efficiency. IEEE Radio Communications. 2004.

[20] K. Berg, H. Ehtamo, M. Uusitalo, C. Wijting. Priority-based classification of spectrum access models. IEEE Communications Magazine. August 2013. Pages 142-147. 2013.

[21] N. Megiddo, C. Papadimitriou. A note on total functions, existencetheorems and complexity. Theoretical Computer Science 81, pages 317–324. 1991.